CAS CS 538. Problem Set 8

Due in class Tuesday, November 20, 2012, *before* the start of lecture

NOTE: This problem set is longer and has more total points that our typical problem sets.

In the two problems below, suppose $\{F_k\}$ is a pseudorandom function family from n-bit inputs to n-bit outputs, with a n-bit key k. Also, let $\circ$ denote concatenation, $\bar{x}$ denote the bit-by-bit negation of x, and 0^n denote the string of n zeroes.

Problem 1. (40 points, 10 for each part) We are trying to construct a function family with $2n$-bit outputs out of $\{F_k\}$. Which of the following are pseudorandom function families? Prove your answers.

(a) $F_k^1(x) = F_k(0^n) \circ F_k(x)$.

(b) $F_k^2(x) = F_k(x) \circ F_k(\bar{x})$.

(c) $F_k^3(x) = F_{0^n}(x) \circ F_k(x)$.

(d) $F_k^4(x) = G(F_k(x))$, where G is a length-doubdling PRG.

Problem 2. (20 points, 10 for each part)

(a) Consider the following function family $\{G_s\}$: if $s = 0^k$, output 0^k; else output $F_s(x)$. Show that it is pseudorandom (a simple reduction works here).

(b) Consider the family $H_s(x) = G_x(s)$. Show that it is **not** pseudorandom.

Thus, swapping the seed and the input in a PRF does not always result in a PRF!

Problem 3. (30 points)

(a) (10 points) Show that two-round Luby-Rackoff is not a pseudorandom permutation by exhibiting a two-query distinguisher. Hint: ask for two queries on the same R and different Ls.

(b) (20 points) Show that three-round Luby-Rackoff is not super pseudorandom. Namely, exhibit a three-query distinguisher (two forward queries followed by one reverse query), and explain why it distinguishes with high probability. (Hint: first, use what you did the previous part. Notice that you can also XOR a predictable value into S with a reverse query. Using this, you can get S in a reverse query to be equal to S in a different forward query. You will thus get a non-random-looking relationship among the inputs and outputs.)

Problem 4. (40 points) In this problem, we return to the random oracle model to build CCA2 secure public-key encryption. Recall the scheme from Problem Set 7 Problem 1: for a TDP family $\{f_i : D_i \to D_i\}$ and random oracle H, Enc(PK, m) picks a random $r \in D_i$, computes $y = f_i(r)$, $p = H(r)$, $d = p \oplus m$, and outputs $c = (y, d)$. You proved that this scheme is CPA secure. Now let G

be another, independent random oracle[1]. Simply augment the above scheme by adding $\sigma = G(m, r)$ to the ciphertext. Upon decryption, check that σ value is correct and, if not, output $\perp$. Prove that this scheme is CCA2 secure. Hint: the proof is essentially the same as in PS 7 problem 1 (feel free to use the solutions – no need to copy them, you can just say "this part is the same as..."), with one wrinkle: we now need to answer to CCA queries of the form (y, d, σ) from the adversary. How can we answer them when we don't know the secret key and therefore can't decrypt? In order to show that we actually can answer them, first prove

(a) unless the adversary queries G on the pair $(f_i^{-1}(y), d \oplus H(f_i^{-1}(y)))$, the CCA query (y, d, σ) will return $\perp$ with all but negligible probability.

(b) Now complete the reduction.

[1]Note that two random oracles can be easily built out of a single one: just prepend a 0 or a 1 to the input depending on which random oracle you want to use